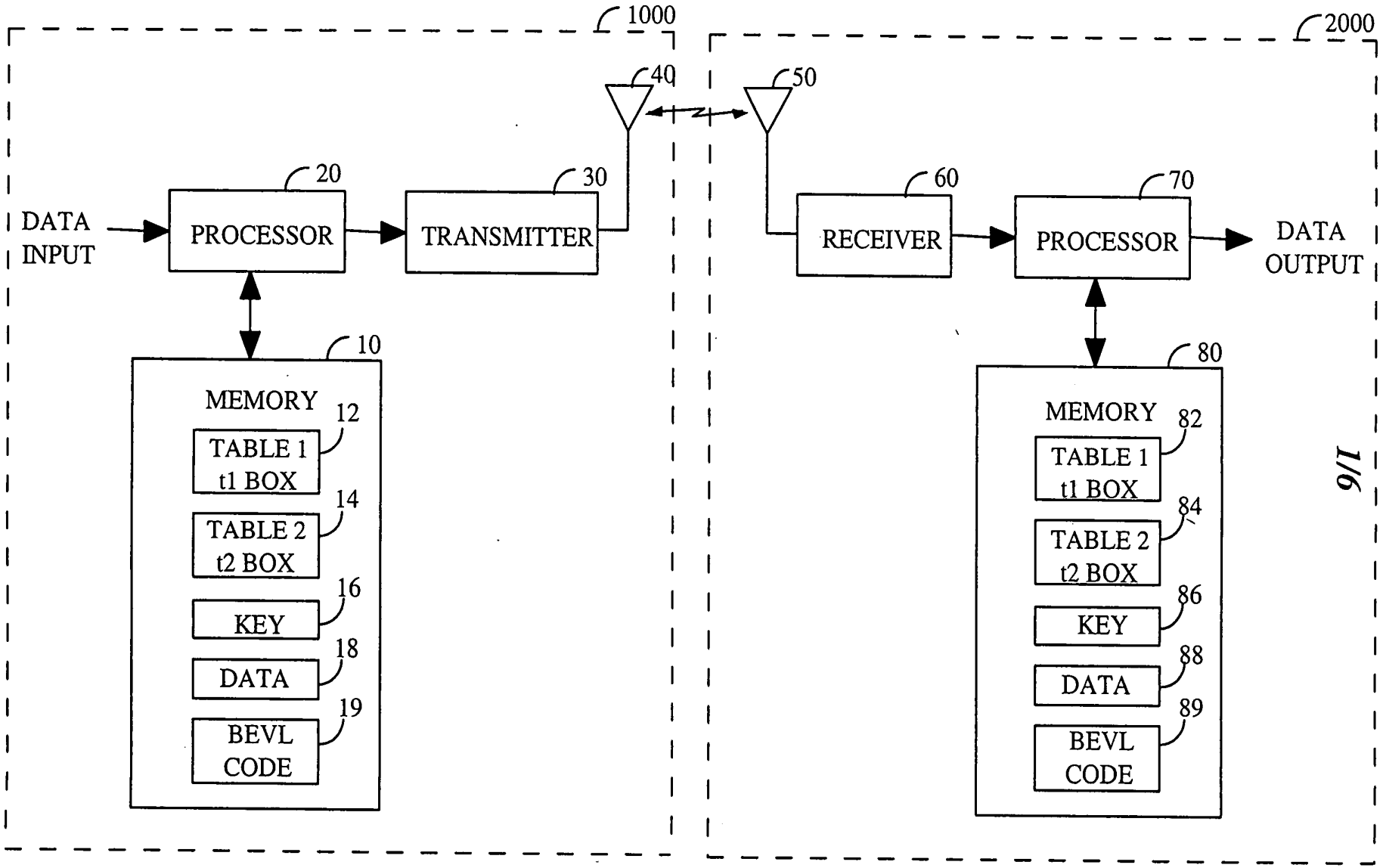
1000
2000
40
50
20
30
60
70
DATA INPUT
PROCESSOR
TRANSMITTER
RECEIVER
PROCESSOR
DATA OUTPUT
10
MEMORY
12
TABLE 1 t1 BOX
14
TABLE 2 t2 BOX
16
KEY
18
DATA
19
BEVL CODE
80
MEMORY
82
TABLE 1 t1 BOX
84
TABLE 2 t2 BOX
86
KEY
88
DATA
89
BEVL CODE
9/I

FIG. 1

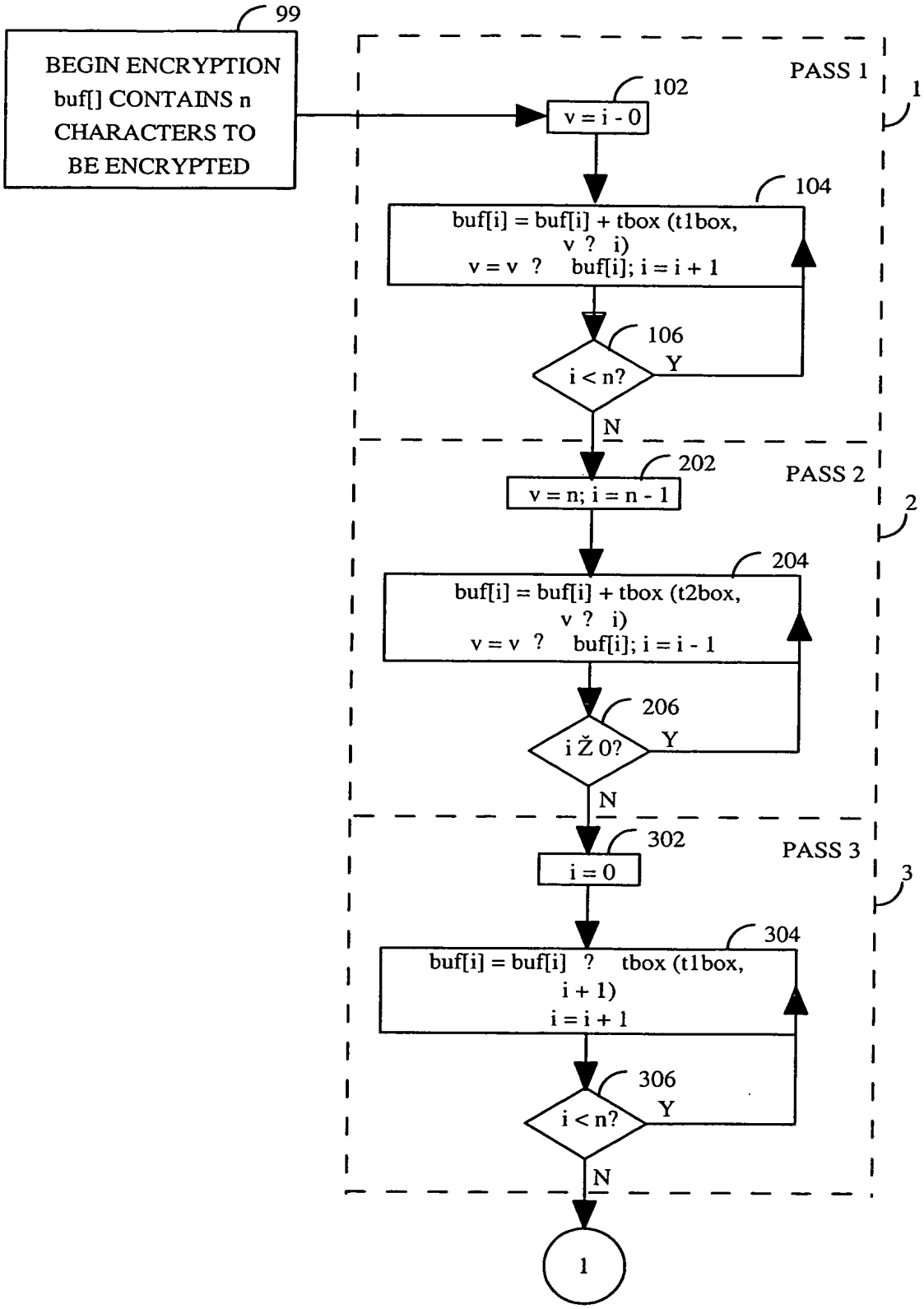
99
BEGIN ENCRYPTION
buf[] CONTAINS n
CHARACTERS TO
BE ENCRYPTED
PASS 1
1
102
v = i - 0
104
buf[i] = buf[i] + tbox (t1box,
v ? i)
v = v ? buf[i]; i = i + 1
106
Y
i < n?
N
PASS 2
2
202
v = n; i = n - 1
204
buf[i] = buf[i] + tbox (t2box,
v ? i)
v = v ? buf[i]; i = i - 1
206
Y
i Ž 0?
N
PASS 3
3
302
i = 0
304
buf[i] = buf[i] ? tbox (t1box,
i + 1)
i = i + 1
306
Y
i < n?
N
1

FIG. 2A

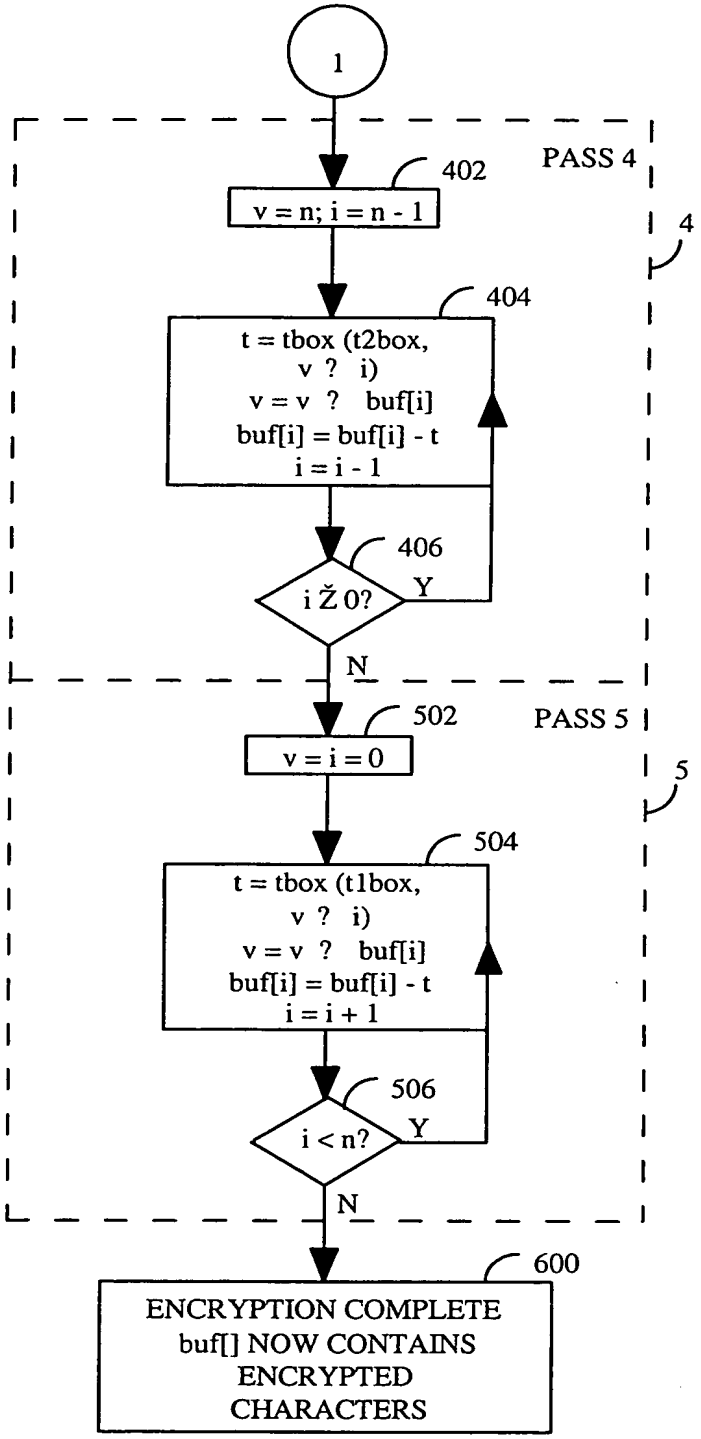
1
PASS 4
402
v = n; i = n - 1
4
404
t = tbox (t2box,
v ? i)
v = v ? buf[i]
buf[i] = buf[i] - t
i = i - 1
406
i Ž 0?
Y
N
PASS 5
502
v = i = 0
5
504
t = tbox (t1box,
v ? i)
v = v ? buf[i]
buf[i] = buf[i] - t
i = i + 1
506
i < n?
Y
N
600
ENCRYPTION COMPLETE
buf[] NOW CONTAINS
ENCRYPTED
CHARACTERS

FIG. 2B

```
/* BEVL - BLOCK ENCRYPTION VARIABLE LENGTH */
unsigned char key[8];

unsigned char tlbox[256] = {
#include "tl.h"
};
unsigned char t2box[256] = {
#include "t2.h"
};

unsigned char
tbox (unsigned char B[256], unsigned char tv)
{
        unsigned char           t;

        t = B[tv ^ key[0]];
        t = B[(t + key[1]) & 0xFF];
        t = B[t ^ key[2]];
        t = B[t + key[3]) & 0xFF];
        t = B[t ^ key[4]];
        t = B[t + key [5]) & 0xFF];
        t = B[t ^ key[6]];
        t = B[t + key[7]) & 0xff];
        t = B[t ^ key[0]];
        return t;
}

void
bev1(unsigned char *buf, int n)
{
        int                     i;
        unsigned char           t, v;

        for (v = i = 0; i < n; ++i) {
                buf[i] += tbox(t1box, v^i);
                v^= buf[i];
        }
        for (v = i = n; --i>= 0;) {
                buf[i] += tbox(t2box, v^i);
                v^= buf[i];
        }
        for (i = 0; i <n; ++i) {
                buf[i]^=tbox(t1box, i + 1);
        }
        for (v = i = n; --i >= 0;) {
                t = tbox(t2box, v^i);
                v^=buf[i];
                buf[i]-=t;
```

FIG. 3

t1.h

```
0xbc, 0xa6, 0x39, 0xd5, 0x02, 0xd2, 0x81, 0xce,
0xf9, 0x38, 0x9d, 0x0a, 0x09, 0xc3, 0x94, 0xf6,
0xe9, 0x92, 0xda, 0xbe, 0x1b, 0x2f, 0x89, 0x3c,
0x48, 0x96, 0x3e, 0x4a, 0xfc, 0xfe, 0xe3, 0x0e,
0xb3, 0x50, 0x7a, 0xc9, 0x40, 0xff, 0x2c, 0xba,
0x47, 0x2e, 0x30, 0x9e, 0x12, 0xa9, 0xa1, 0xd8,
0xb5, 0xa8, 0x1d, 0x04, 0x33, 0xb0, 0xef, 0x95,
0xa3, 0xed, 0x32, 0x59, 0x68, 0x2b, 0xe1, 0xf3,
0x2a, 0xb8, 0x1c, 0x21, 0x13, 0xb2, 0x60, 0x66,
0x9c, 0x74, 0x5b, 0xb1, 0x18, 0xec, 0x77, 0xd0,
0x1f, 0xc2, 0x3f, 0xf8, 0x1e, 0x93, 0xae, 0x8c,
0x23, 0xc6, 0x3d, 0x5e, 0xe7, 0x15, 0xc1, 0x62,
0xa7, 0x8f, 0xdd, 0x90, 0xf1, 0xb4, 0x34, 0x3a,
0x2d, 0xf4, 0xe8, 0xcf, 0x53, 0xfa, 0x25, 0x69,
0x7b, 0x36, 0xad, 0x9a, 0x7c, 0x58, 0x05, 0x9b,
0xf7, 0x49, 0xa2, 0x7f, 0xdc, 0xf5, 0xcb, 0xd6,
0xdb, 0x8a, 0xc5, 0x20, 0x27, 0xd7, 0x8b, 0x14,
0xd9, 0x16, 0xc0, 0x28, 0xaa, 0x57, 0x78, 0x0d,
0x26, 0x7e, 0x87, 0x4c, 0x24, 0xeb, 0x99, 0x4e,
0x64, 0x71, 0x6f, 0xe4, 0x84, 0x46, 0xb7, 0x17,
0x76, 0x88, 0xbb, 0xe0, 0x3b, 0xca, 0x8e, 0x83,
0x5a, 0x43, 0x4d, 0xa5, 0x6c, 0x6a, 0x63, 0x82,
0xac, 0x0c, 0xfd, 0xab, 0x9f, 0x1a, 0x55, 0x41,
0x80, 0x22, 0x00, 0xc8, 0x65, 0x08, 0x73, 0x6b,
0xcc, 0x67, 0x56, 0xdf, 0x85, 0x42, 0x5f, 0xe2,
0xbd, 0x91, 0x44, 0x07, 0x86, 0x7d, 0x75, 0xb9,
0xb6, 0xe6, 0x4b, 0xc7, 0x61, 0xee, 0xf0, 0xf2,
0xa0, 0xaf, 0x31, 0x06, 0xfb, 0x10, 0x72, 0x4f,
0x97, 0xe5, 0xbf, 0x5d, 0xd1, 0xde, 0xc4, 0x5c,
0xcd, 0x37, 0xea, 0x11, 0x79, 0x6e, 0x54, 0x0b,
0x45, 0xa4, 0x8d, 0x01, 0x03, 0x51, 0x35, 0x70,
0x52, 0x98, 0xd4, 0x19, 0x6d, 0xd3, 0x0f, 0x29,
```

FIG. 4

t2.h

```
0xde, 0x75, 0xe3, 0x3e, 0x66, 0x2c, 0xfc, 0x77,
0x43, 0x81, 0xdb, 0x10, 0xa3, 0x9a, 0xcd, 0x7e,
0x8d, 0x7b, 0x64, 0x6e, 0x52, 0x0c, 0x15, 0x23,
0x95, 0x65, 0xd1, 0xb4, 0x5d, 0x73, 0x63, 0xac,
0x59, 0xd0, 0xc3, 0xf6, 0xf3, 0xaf, 0x61, 0x89,
0x53, 0x5b, 0x79, 0x83, 0x9f, 0x08, 0x0e, 0xa4,
0xd6, 0x1c, 0xe4, 0x93, 0x3f, 0xe9, 0x44, 0xad,
0xa9, 0x01, 0x99, 0x6c, 0xea, 0xc6, 0x49, 0x92,
0xff, 0x58, 0xbb, 0x28, 0x19, 0xdd, 0xa0, 0xbc,
0xe6, 0x0d, 0xca, 0x88, 0x25, 0x06, 0xd8, 0x2b,
0xbd, 0xba, 0x17, 0xcf, 0xc7, 0xd2, 0xd9, 0x46,
0x60, 0x7f, 0xe8, 0x48, 0xd3, 0x4b, 0x7c, 0x0b,
0xbe, 0xa6, 0xc5, 0xb6, 0x70, 0x5a, 0x7d, 0x37,
0x8c, 0x34, 0x9c, 0x6d, 0x39, 0x21, 0x3b, 0x1d,
0xcb, 0x09, 0x24, 0x1a, 0x5e, 0x1e, 0x14, 0xb2,
0x7a, 0x0a, 0x4a, 0x85, 0xb1, 0x8b, 0x4e, 0x4f,
0xd7, 0xf5, 0xda, 0x6b, 0x87, 0x96, 0x55, 0x32,
0x3a, 0xb9, 0x47, 0x3c, 0x71, 0xdc, 0x74, 0xb5,
0xed, 0xfd, 0x62, 0x80, 0x03, 0x2e, 0x98, 0xc8,
0xa2, 0x90, 0x67, 0x84, 0xab, 0xf1, 0xa7, 0xee,
0xe7, 0x50, 0xf8, 0xd5, 0x45, 0xae, 0x00, 0x9e,
0x07, 0x5f, 0x22, 0xf7, 0xc2, 0x4d, 0x8f, 0x4c,
0x82, 0x57, 0x68, 0xc9, 0x04, 0xfe, 0x11, 0xfa,
0x91, 0x31, 0x13, 0xef, 0x8e, 0x12, 0x94, 0xf2,
0x38, 0xec, 0x30, 0x16, 0x76, 0x2a, 0xe0, 0x29,
0x9b, 0xf4, 0xf9, 0x33, 0x6a, 0x78, 0xbf, 0x69,
0x51, 0xe2, 0xe5, 0x41, 0xb7, 0xeb, 0x26, 0xb0,
0x1f, 0x35, 0x6f, 0xa8, 0x9d, 0x3d, 0xaa, 0xb8,
0xc1, 0x27, 0x18, 0xdf, 0xe1, 0x42, 0x54, 0xcc,
0x1b, 0x72, 0x2d, 0xa1, 0xc4, 0x05, 0x40, 0x5c,
0xb3, 0xa5, 0x97, 0xc0, 0x36, 0x20, 0x02, 0x2f,
0xce, 0xf0, 0x0f, 0xd4, 0x8a, 0x56, 0xfb, 0x86,
```

FIG. 5